ASSIGNMENT-5

1. Software user input to hardware action

**Code:**

**while** (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

HAL\_UART\_Receive(&huart1, UART1\_rxBuffer,12,10000);

HAL\_UART\_Transmit(&huart1,UART1\_rxBuffer,**sizeof**(UART1\_rxBuffer),10000);

**if**(\*UART1\_rxBuffer=='1'){

HAL\_GPIO\_WritePin(led\_GPIO\_Port, led\_Pin, 1);

HAL\_GPIO\_WritePin(led2\_GPIO\_Port, led2\_Pin, 0);

HAL\_UART\_Transmit(&huart1, (uint8\_t \*)"led1 on!", 10,10000);

}

**if**(\*UART1\_rxBuffer=='2') {

HAL\_GPIO\_WritePin(led\_GPIO\_Port, led\_Pin, 1);

HAL\_GPIO\_WritePin(led2\_GPIO\_Port, led2\_Pin, 1);

HAL\_UART\_Transmit(&huart1, (uint8\_t \*)"led1 and led2 on!", 30,10000);

}

**if**(\*UART1\_rxBuffer=='3'){

HAL\_GPIO\_WritePin(led\_GPIO\_Port, led\_Pin, 0);

HAL\_GPIO\_WritePin(led2\_GPIO\_Port, led2\_Pin, 0);

HAL\_UART\_Transmit(&huart1, (uint8\_t \*)"led1 and led2 off!", 20,10000);

}

**if**(\*UART1\_rxBuffer=='4'){

**while**(1){

HAL\_GPIO\_WritePin(led\_GPIO\_Port, led\_Pin, 1);

HAL\_Delay(1000);

HAL\_GPIO\_WritePin(led\_GPIO\_Port, led\_Pin, 0);

HAL\_GPIO\_WritePin(led2\_GPIO\_Port, led2\_Pin, 1);

HAL\_Delay(1000);

HAL\_GPIO\_WritePin(led2\_GPIO\_Port, led2\_Pin, 0);

HAL\_UART\_Transmit(&huart1, (uint8\_t \*)"Invalid Input!", 20,10000);

}

}

HAL\_UART\_Transmit(&huart1, UART1\_endBuffer, **sizeof**(UART1\_endBuffer),10000);

}

**Configuration:**

**Graphical user interface, application

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

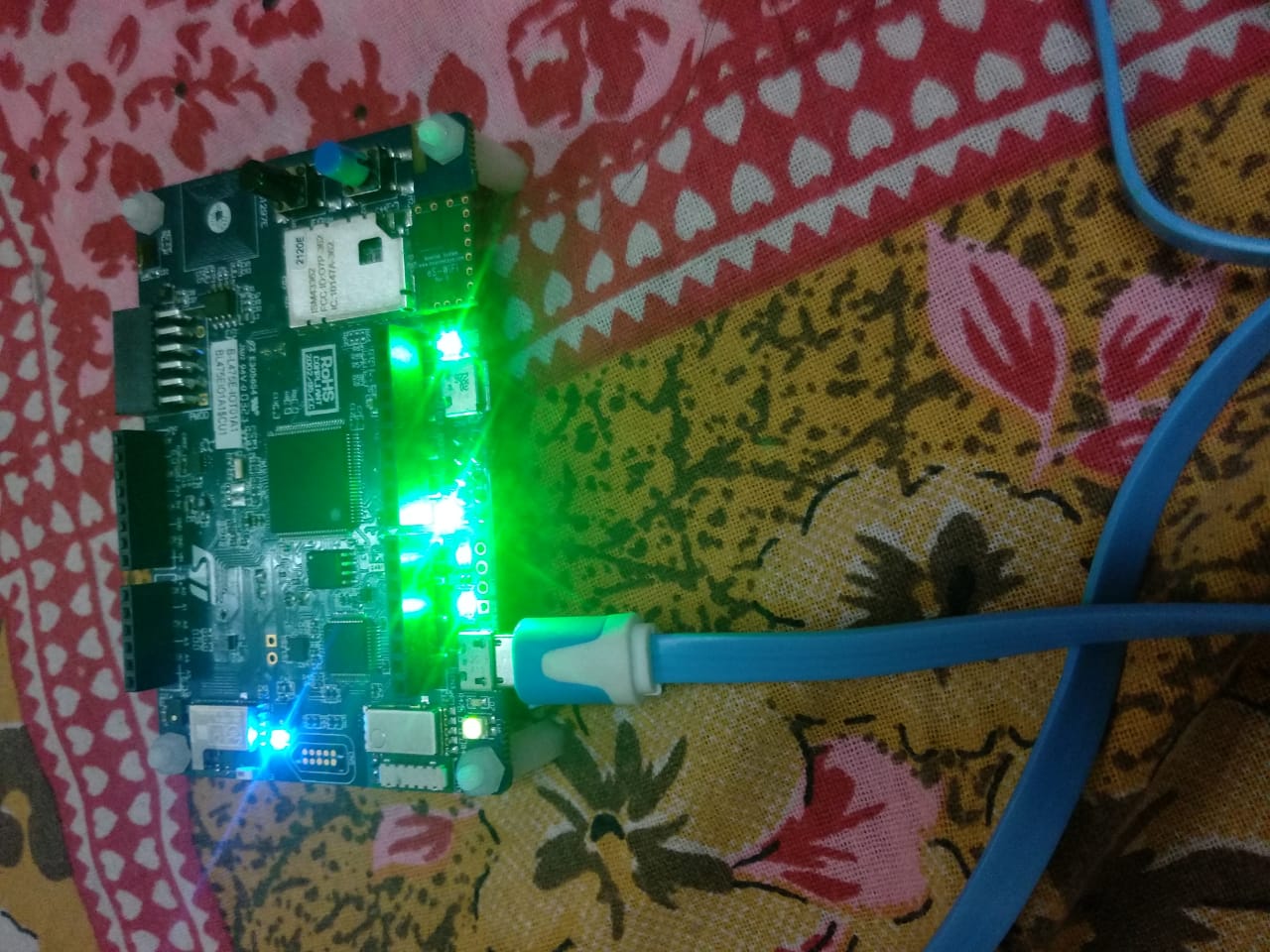
Ouput:

From terra terminal

Graphical user interface, text, application

Description automatically generated

Output on MCU :



1. Morse code:

**while** (1)

{

/\* USER CODE END WHILE \*/

/\* USER CODE BEGIN 3 \*/

//HAL\_UART\_Transmit(&huart1,input , sizeof(input), 10000);

//HAL\_UART\_Transmit(&huart1,"enter name in upper case ", 50, 10000);

//HAL\_Delay(6000);

HAL\_UART\_Receive\_DMA(&huart1, input, 5000);

//HAL\_UART\_Transmit(&huart1,input , sizeof(input), 10000);

//HAL\_UART\_Transmit(&huart1,"Entercharacter" , 40, 10000);

HAL\_Delay(5000);

ledMorseAlphabet(input[i]);

/\*space(); // 3 morse unit time space between 2 initials

space();

space();\*/

i=i+1;

}

**void** **HAL\_UART\_RxCpltCallback**(UART\_HandleTypeDef \*huart) {

HAL\_UART\_Receive\_DMA(&huart1, input, 10000);

}

**void** **dot**() {

HAL\_UART\_Transmit(&huart1, " . ", 3, 1000);//Serial output to show user, current state

HAL\_GPIO\_WritePin(led\_morse\_GPIO\_Port, led\_morse\_Pin, *GPIO\_PIN\_SET*);

HAL\_Delay(1000);

HAL\_GPIO\_WritePin(led\_morse\_GPIO\_Port, led\_morse\_Pin, *GPIO\_PIN\_RESET*);

HAL\_Delay(1000);

**return**;

}

**void** **dash**() {

HAL\_UART\_Transmit(&huart1, " - ", 3, 1000); //Serial output to show user, current state

HAL\_GPIO\_WritePin(led\_morse\_GPIO\_Port, led\_morse\_Pin, *GPIO\_PIN\_SET*);

HAL\_Delay(3000);

HAL\_GPIO\_WritePin(led\_morse\_GPIO\_Port, led\_morse\_Pin, *GPIO\_PIN\_RESET*);

HAL\_Delay(1000);

**return**;

}

**void** **space**() {

HAL\_Delay(1000);

HAL\_UART\_Transmit(&huart1, " ", 3, 1000); //Serial output to show user, current state

**return**;

}

**void** **ledMorseAlphabet**(uint8\_t input) {

**switch** (input) {

**case** 'A':

dot();space();dash();

**break**;

**case** 'B':

dash();space();dot();space();dot();space();dot();

**break**;

**case** 'C':

dash();space();dot();space();dash();space();dot();

**break**;

**case** 'D':

dash();space();dot();space();dot();space();

**break**;

**case** 'E':

dot();

**break**;

**case** 'F':

dot();space();dot();space();dash();space();dot();

**break**;

**case** 'G':

dash();space();dash();space();dot();

**break**;

**case** 'H':

dot();space();dot();space();dot();space();dot();

**break**;

**case** 'I':

dot();space();dot();

**break**;

**case** 'J':

dot();space();dash();space();dash();space();dash();

**break**;

**case** 'K':

dash();space();dot();space();dash();

**break**;

**case** 'L':

dot();space();dash();space();dot();space();dot();

**break**;

**case** 'M':

dash();space();dash();

**case** 'N':

dash();space();dot();

**case** 'O':

dash();space();dash();space();dash();space();

**break**;

**case** 'P':

dot();space();dash();space();dash();space();dot();

**break**;

**case** 'Q':

dash();space();dash();space();dot();space();dash();

**break**;

**case** 'R':

dot();space();dash();space();dot();

**break**;

**case** 'S':

dot();space();dot();space();dot();

**break**;

**case** 'T':

dash();

**break**;

**case** 'U':

dot();space();dot();space();dash();

**break**;

**case** 'V':

dot();space();dot();space();dot();space();dash();

**break**;

**case** 'W':

dot();space();dash();space();dash();

**break**;

**case** 'X':

dash();space();dot();space();dot();space();dash();

**break**;

**case** 'Y':

dash();space();dot();space();dash();space();dash();

**break**;

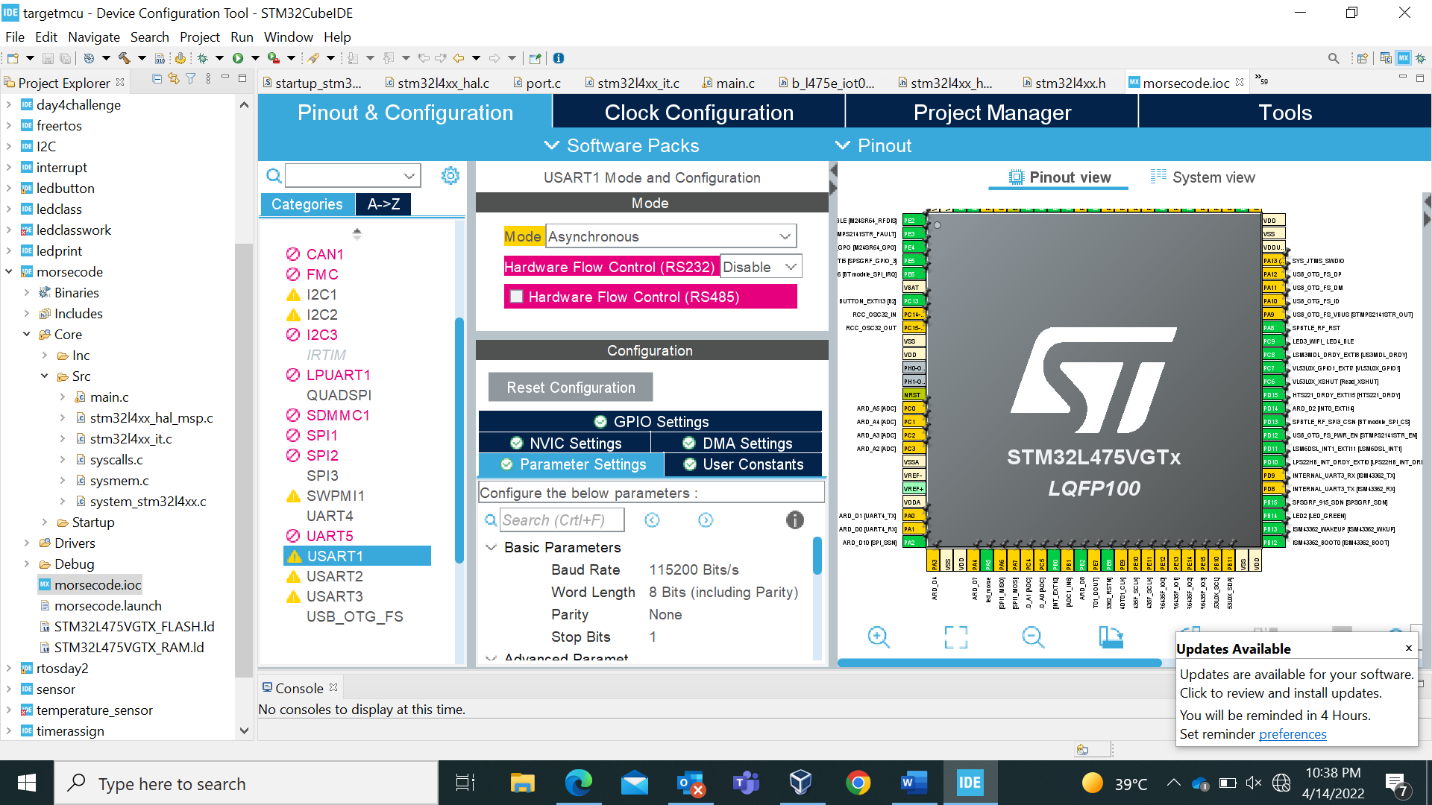
**case** 'Z':

dash();space();dash();space();dot();space();dot();

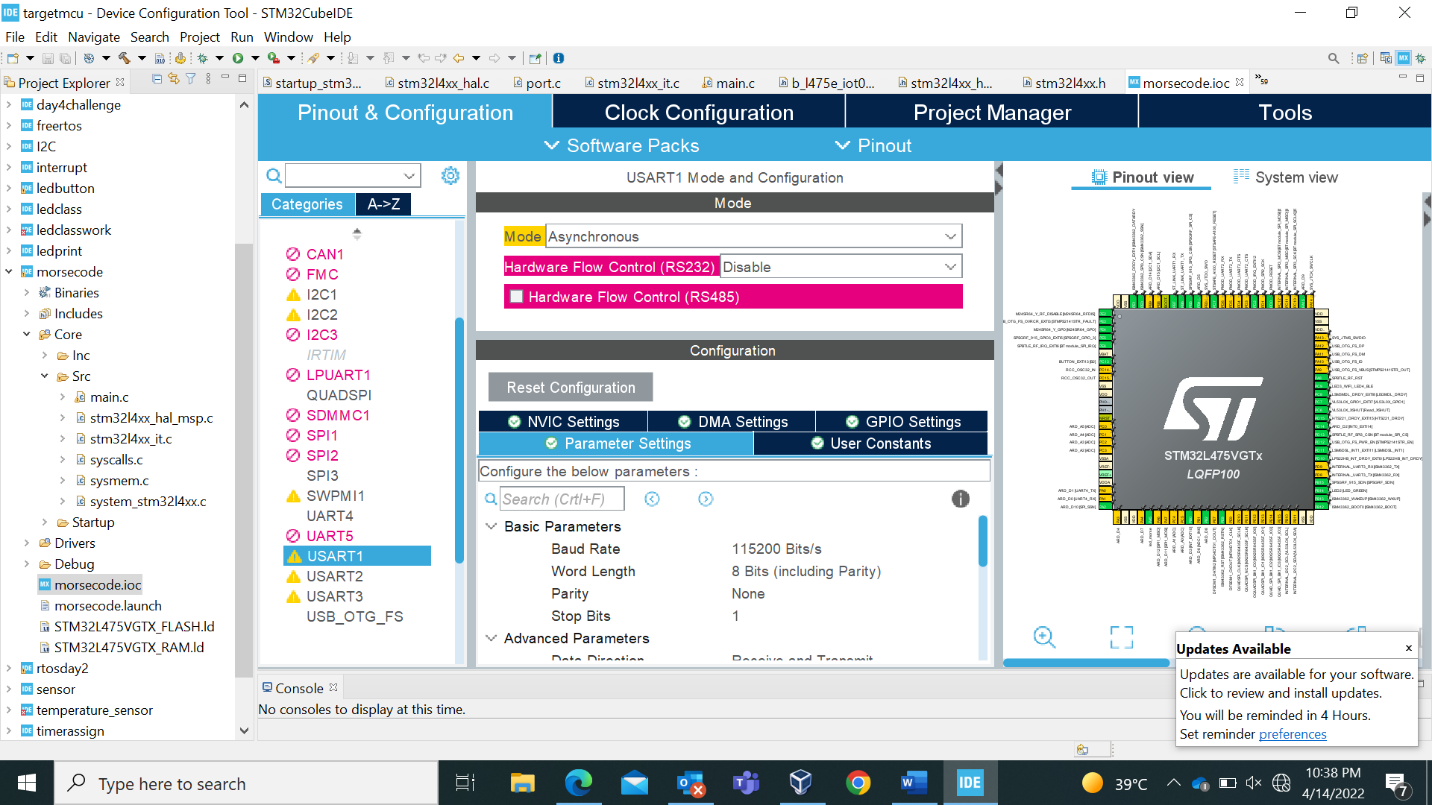
**break**;

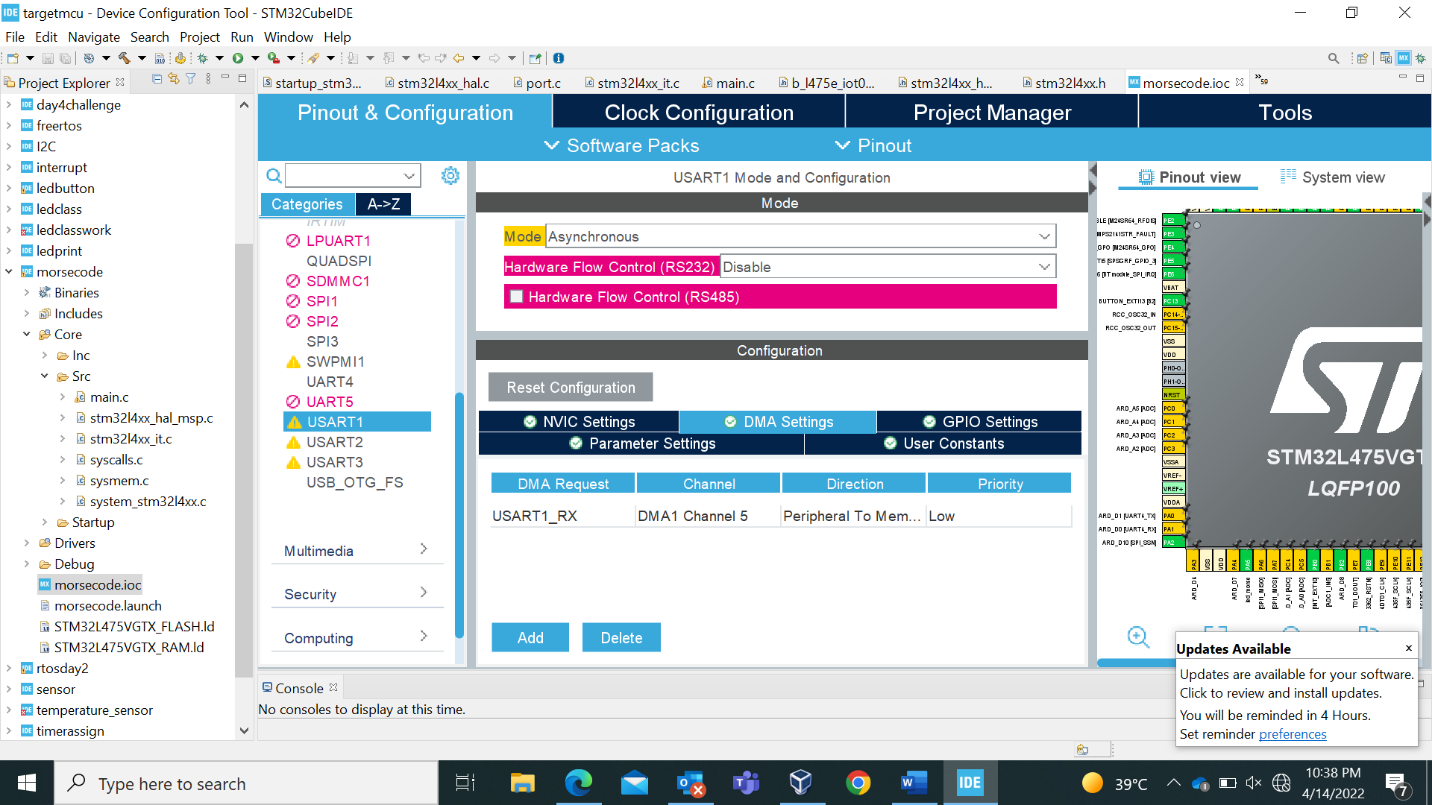
Configurations:

Pin configuration:

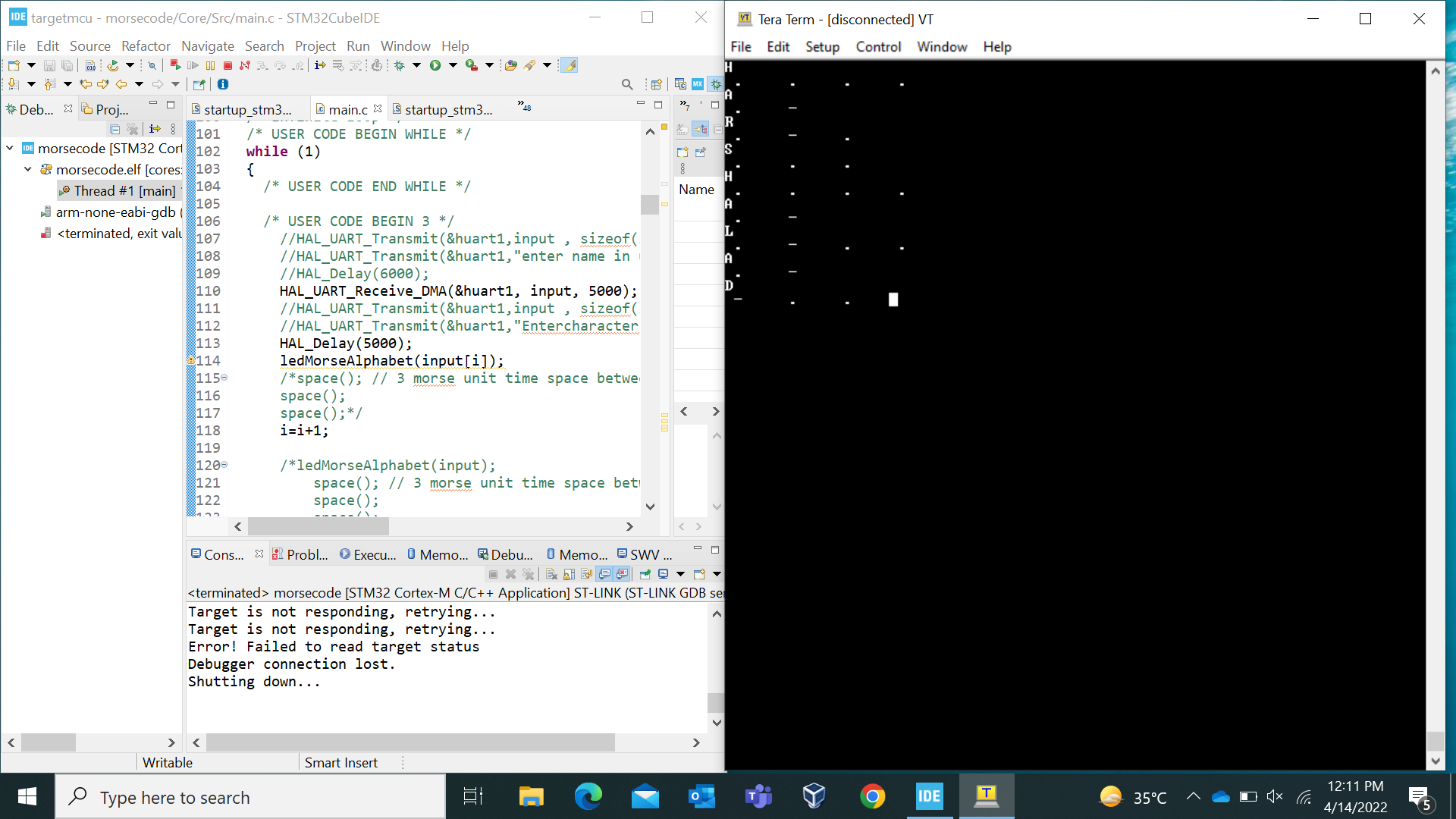


USART DMA mode configuration:





Output:



Morse code output on LED: for dash led will be on for 3 seconds and for dot led will be on for 1 second

